TIME SHIFTING AIR TRAFFIC DATA FOR QUANTITATIVE EVALUATION OF A CONFLICT PROBE

Mike M. Paglione*
FAA William J. Hughes Technical Center
tlantic City International Airport, NJ, 08405

Robert D. Oaks[†] and J. Scott Summerill[‡]
Veridian IT Services, Inc.
Egg Harbor Township, NJ 08234

Abstract

A conflict probe is an air traffic management decision support tool that predicts aircraft-to-aircraft and aircraft-to-airspace conflicts. In order to achieve the confidence of the air traffic controllers who are provided this tool, a conflict probe must accurately predict these conflicts. This paper discusses how a conflict probe's quantitative accuracy requirements can be tested using hypothesis testing techniques. The paper also asserts that air traffic scenarios based on recorded field data are essential to the evaluation of a conflict probe and states that time shifting these scenarios can create data samples necessary to perform the hypothesis testing. This paper then compares three time shifting techniques: time compression, random time adjustment, and an implementation of a genetic algorithm.

Introduction

A ground-based conflict probe (CP) is a decision support tool that provides the air traffic controller with predictions of conflicts (i.e., loss of minimum separation between aircraft) for a parameter time (e.g., 2 to 20 minutes) into the future. At a minimum, a conflict probe predicts the flight path of an aircraft, continuously monitors that flight path from current aircraft position information, and probes for conflicts with other aircraft and possibly incursions into restricted airspace. Some versions also assist the controller in resolving the predicted conflicts, and with alternative route planning in response to user requests.

There are several different types of conflict probes, differentiated mainly by their time domain and

---

* Conflict Probe Assessment Team Lead, Simulation and Analysis Branch, ACB-330; *mike.paglione@faa.gov*
† Senior Computer Systems Analyst and member of AIAA; *rdoaks@acm.org*
‡ Applications Engineer; *scott.ctr.summerill@faa.gov*

application in the National Airspace System (NAS). Some focus on more near-term or tactical conflict predictions (2-3 minutes), while others predict longer-term (up to 20 minutes) conflicts between aircraft. Some versions are adapted for the radar air traffic controller who is responsible for aircraft separation. Others are designed for the associate or planning controller and provide separate displays, which are indirectly accessible by the radar controller. These longer-term conflict probes support the reduction of restrictions by aiding the controller in the strategic planning of aircraft separation management. All make predictions of the future path of an aircraft, and thus are subject to some uncertainty.

As described in [1], the Federal Aviation Administration (FAA) deployment of one such CP requires quantitative accuracy measurement to ensure the application performs within a specified level of accuracy. The primary quantitative measurement of a CP's overall accuracy is the evaluation of its conflict predictions. This is consistent with the probe's central goal of detecting conflicts. As presented in [2], [3], and [4], conflict prediction accuracy quantifies the two fundamental error probabilities of a CP: Missed and False Alerts. A Missed Alert occurs when a CP fails to predict a conflict. A False Alert occurs when a CP predicts a conflict that does not actually occur.

To measure the Missed Alerts, a significant number of CP predictions of aircraft-to-aircraft conflicts (i.e. loss of legal separation between the aircraft) need to be examined. To measure False Alert prediction errors, a significant number of non-conflict predictions or encounters between aircraft need to be examined. Since air traffic controllers separate aircraft to avoid conflicts, these events generally cannot be observed from actual air traffic operations. Non-conflicts or encounters between aircraft do occur in actual air traffic operations but often not at the required levels needed for accuracy measurement. One approach would be to generate the events by simulation. However, air traffic messages contain errors and inconsistencies (e.g. surveillance

radar tracking error, lack of flight intent in the form of missing or timely aircraft controller directives, navigation errors, etc.). A more realistic method is to use recorded air traffic data. In [3], recorded traffic data was utilized and the definition of conflicts was re-defined. Another approach is time shifting the recorded air traffic to induce conflicts and encounters. The focus of this paper will be on time shifting recorded air traffic data to induce encounters only. Future research will address both conflict and encounter events in an analogous manner.

In this paper the FAA's Conflict Probe Assessment Team (CPAT) first presents a quantitative approach to measuring the conflict prediction errors. Next, several methods of time shifting recorded air traffic data are introduced. Finally, the various time shifting techniques are compared using a sample of air traffic data from Memphis Air Route Traffic Control Center (ARTCC).

Quantitative Requirements Testing

As presented in [1], a CP requires quantitative accuracy measurement to ensure the application performs within a specified level of accuracy. This is accomplished by performing accuracy tests on the CP that bound key system level requirements, such as the probability of Missed Alert or False Alert events. The test's probability of a False Alert ($\hat{p}$) is defined in the following Equation 1. The probability of Missed Alert is similarly quantified as a proportion of conflicts as defined in [1].

$$\hat{p} = \text{probability of False Alert} = \frac{f}{n} \qquad (1)$$

where

f = number of False Alert encounters
n = total number of encounters

Expressed in Equation 2, the quantitative measurement tests the CP's probability of False Alert in Equation 1 to ensure it is less than or equal to the required probability of False Alert.

$$\hat{p} \le P_r \qquad (2)$$

where

$\hat{p}$ = test probability of False Alert
P_r = required probability of False Alert

As discussed in the previous section, recorded traffic data is utilized to provide n realistic encounters. Since $\hat{p}$ is sample estimate of the true probability of False Alert, a statistical inference method, called hypothesis testing, is employed to provide statistical evidence that the CP actually passed or failed the test. The next section presents the hypothesis testing approach and provides guidance to scenario developers in the quantity of encounters needed.

Hypothesis Testing and Required Sample Size

Hypothesis testing is an inferential technique used to make a broad claim on the value of some population parameter or characteristic. The practitioner will hypothesize a value for the population parameter and then use a statistic from the test population to support or reject the premise. It is also possible given an estimate of the unknown parameter to determine the number of sample observations required to provide the most powerful test. This was the application used in the current study where the proportion of aircraft encounters was determined and the corresponding number of encounters required to achieve a fixed level of statistical significance was calculated. A detailed discussion of hypothesis testing can be found in [5].

In hypothesis testing two complementary statements are postulated regarding the true but unknown value for some population parameter. In this study, the null and alternative hypothesis for a one-sided test can be stated as,

H_0: $P <= P_r$, the true (and unknown) population parameter is less than or equals a fixed value P_r

H_1: $P > P_r$, the population parameter is actually greater than value P_r

Here H_0 and H_1 state the null and alternative hypothesis, the variable P is an unknown population parameter and P_r is the hypothesized value for that parameter. In this example, P_r represents the required probability of False Alert, and P represents the true and unknown probability of False Alert. The method assumes that both the sampling distribution of the test statistic and that of the population parameter are normally distributed with equal variances. Figure 1 shows the test setting.
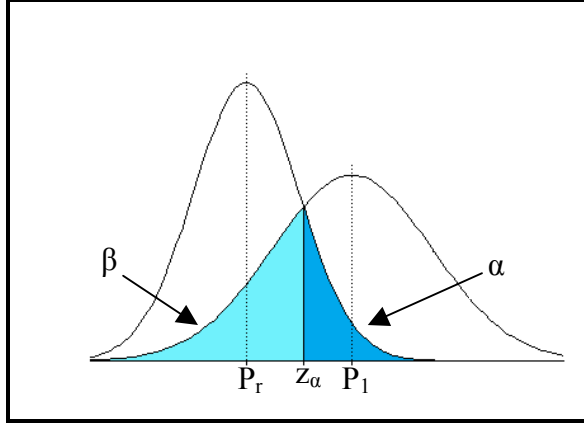
Figure 1: Hypothesized vs. True Distributions

The figure shows two normal curves. The P_r value demarks the mean (and also the median and mode) of a normal curve for the hypothesized population. The P_l value demarks the mean of a normal curve for the true population. The true population is actually fixed relative P_r and results in some degree of overlap in the tail regions between the two curves. This overlap is labeled as α and β and will be further defined below. Note that the above figure is shows unequal variances (standard deviations). The test assumes equal variance in both populations and will be further explained below.

Figure 1 further identifies a vertical line labeled z_α that intersects the two population curves, a region to the right of z_α under the P_r curve labeled as α and a region to the left of z_α under the P_l curve labeled as β. The z_α is a number of standard deviations to the right of the hypothesized mean of the P_r curve that captures a predetermined probability in the tail region. The practioner will select some probability (known as α or the critical region) in the tail where a sample is considered so unlikely that observing such a sample will support the alternative hypothesis. For this selected probability the corresponding z_α is then a look-up value from a table for the standard normal curve. For a given z_α and fixed sample size, β is then completely determined and represents the region under the P_l curve to the left of z_α where a sample will erroneously provide evidence supporting the null hypothesis. Equation 3 shows calculation of the β probability.

$$\beta\left(p_1\right) = \Phi\left[\frac{p_r + z_\alpha\sqrt{\dfrac{p_r\left(1-p_r\right)}{n}} - p_1}{\sqrt{\dfrac{p_1\left(1-p_1\right)}{n}}}\right] \quad (3)$$

A conclusion supporting either hypothesis is based on the z-statistic. The z-statistic standardizes the distance between a sample estimate of the true population mean and the hypothesized mean. Equation 4 shows the calculation for the z-statistic where $\hat{p}$ is the sample estimate, p_r is the hypothesized parameter value and n is the sample size. A value for the test statistic falling in the α critical region provides evidence against the null hypothesis. For this study it provides evidence that the CP failed the test and the true number of False Alerts was larger than the requirement.

$$z = \frac{\hat{p} - p_r}{\sqrt{\dfrac{p_r\left(1-p_r\right)}{n}}} \quad (4)$$

Historically the α region has been known as the producer's risk as an erroneous sample may result in rejecting product from a process that is working correctly. The β region has been known as the consumer's risk as a sample may fail to reveal a process that is actually generating defective product. In this study, the two critical regions were held equal to balance the risk to both the consumer (e.g. the FAA) and the producer (e.g. the CP developer).

Another consideration is sample size, which can be considered a balance between sufficiency (large to provide good information on the true population) and practicality (sampling constraints). It should be noted that increasing the sample size is the only means of reducing the β critical region for a fixed level of α. Increasing the sample size has the effect of reducing the standard error which reduces the overlap between distributions. In an earlier paragraph it was noted that the population curves of Figure 1 were illustrated with unequal variances. They were drawn as such to clarify the effect of increased sample size. If both populations originally had a shape (or variance) consistent with that of the P_l curve some increase in the sample size would have the effect of reducing the variance and generate the more peaked shape of the P_r curve. Then increasing sample size with P_r and P_l fixed reduces the overlap in the tail regions between distributions and consequently the β critical region is reduced (assuming a fixed α). In this work the variable of interest was the sample size required to reduce the population variance such that β was equal to a pre-determined, fixed value for α.

The data in this study is dichotomous (0-1, pass-fail, encounter or no encounter). A statistic based on a dichotomous random variable can be modeled as having a Binominal sampling distribution, which for a

large sample size can be approximated using the normal distribution. This normal approximation to the binomial enables the use of Equations 3 and 4 in determining a value for β. In this study, for a pre-determined value of α and β, and a fixed value for the two population means, the required number of observations can be determined as shown in Equation 5. Here the z_α and z_β parameters are the number of standard deviations above and below P_r and P_l population means, respectively, and the remaining variables have been defined above.

$$n = \left[\frac{z_\alpha \sqrt{p_r(1-p_r)} + z_\beta \sqrt{p_1(1-p_1)}}{p_1 - p_r} \right]^2 \quad (5)$$

In this work, n represents the number of encounters required for the accuracy test of the CP given a pre-determined α and β and known or assumed values for the P_r and P_l population means.

Application of the Required Sample Size
Equation 5 can be used to determine the number of scenario hours to be generated (time-shifted recorded traffic data). To demonstrate the methodology for this study, the required probability of False Alert is assumed 0.16 (a reasonable value for a CP from the authors' experience). This is P_r in Equation 2. As suggested in

[1], P_l in Equation 5 can be defined as a multiple of P_r. This is expressed in the following Equation 6.

$$p_1 = \lambda(p_r) \quad (6)$$

where $\lambda > 1$ and $p_1 \leq 1$.

Equations 5 and 6 and the assumed requirement value (P_r) are applied with various values of α and β. This is illustrated in Figure 2. The y-axis in Figure 2 represents the required sample size of encounters, and the x-axis is the multiple of P_r or λ from Equation 6. The figure illustrates the resulting curves formed by α and β probabilities ranging from 0.01 to 0.15. The curves show the test will require very large sample sizes for detecting small differences between the requirement and true False Alert probability (small λ) and reduces significantly as the difference increases (larger λ). Also, larger sample sizes are generally required for smaller α and β probabilities. For this study, an α and β of 0.05 is chosen with a λ of 1.05. This equates to 23,179 required sample encounters. The following section defines the methodology of generating the scenario data to produce these encounters.
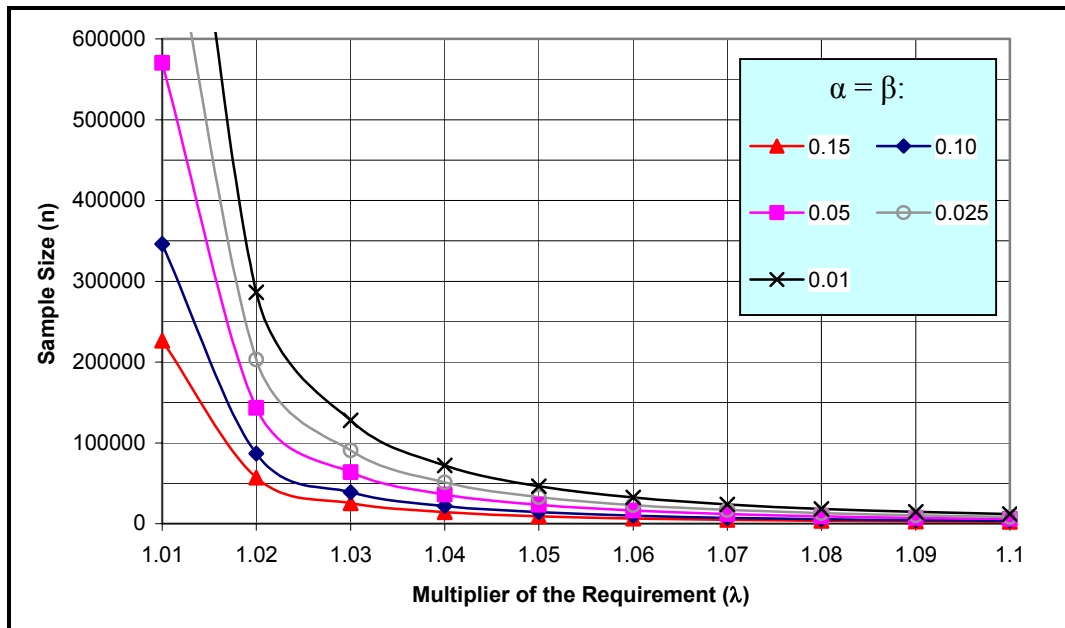

Figure 2: Required Sample Size for Accepted Type I & II Errors

Scenario Time Shifting

CPAT developed the scenario generation process described in [6]. This process consists of three steps: (1) data extraction, where the recorded traffic data is used to populate a set of relational database tables; (2) data modification, where the data in the tables may be manipulated for test purposes; and (3) scenario generation, where scenarios are created based on the traffic data retrieved from the modified database tables.

Track data extracted during the data extraction step preserves real world errors, but generally does not contain the conflicts or the desired encounter distributions to effectively test a conflict probe. Time shifting is a technique used in the data modification step in which flights are moved either forward or backward in time. This causes conflicts and encounters that are not present in the recorded field data, while at the same time retaining the profile of the individual flights.

Figure 3 presents a graphic representation of time shifting. The line on the top of the figure represents the timeline of the recorded data. On this recorded timeline a flight plan message (FP) and numerous track messages (Tk) for a single flight are depicted. During the data extraction step a single start time (labeled *flight.start_time*) is associated with the flight. This start time is the time stamp associated with the first track point for the flight. The times associated with all other events (i.e., the flight plan message and each of the track messages) are retained relative to this start time; this is called flight-centric data as defined in [6].
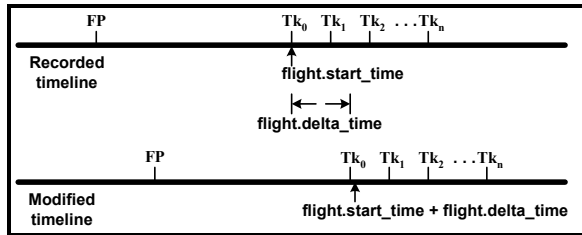

Figure 3: Time Shifting

The lower line in Figure 3 represents the time-shifted timeline that results when a time-shifted value of *flight.delta_time* is applied to the flight. Note that the entire flight is shifted in time but that the relative times between the flight's events remain the same.

Special consideration is given to flights that have the same aircraft id (ACID). It is assumed that these flights are the same aircraft, therefore the same time shift value is applied to each instance.

Time shifting, when done in this manner, causes the individual flights to retain their flight profiles, but the interaction between flights is altered. This causes the conflicts and encounters in the generated scenarios that are not found in the recorded field data.

Three time compression techniques have been used by CPAT: time compression, random time adjustment, and an implementation of the genetic algorithm. These are discussed in the following subsections.

Time Compression

For time compression, a constant (C_m) is multiplied times the difference between the original start time of a flight and a base time that precedes all flights in the scenario. In Figure 4, T_b represents the base time, T_0 represents the flight's start time, and the block labeled *Recorded Track* represents the flight's flight-centric data. With time compression, the flight's original start time, T_0, is changed to T_0' using Equation 7.

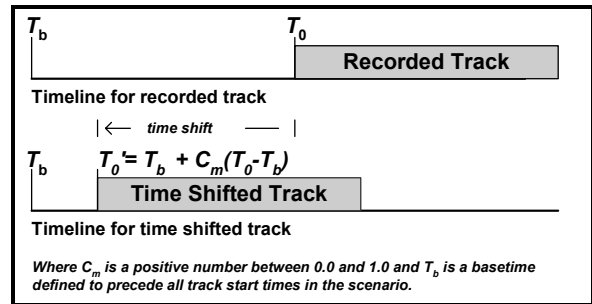$$T_0' = T_b + C_m\left(T_0 - T_b\right) \qquad (7)$$

Figure 4: Time Compression

Note with time compression the flights are always moved earlier in time and that no flight's start time can be shifted earlier than the base time. A time-compressed scenario has a longer duration than a non time-compressed scenario. The following two examples illustrate why this is true:

1. Consider a flight that has multiple flight plans that occur before the track data begins. For example, if the first flight plan occurs shortly after the beginning of the scenario and the track data begins later in the scenario, the track may be shifted earlier a significant amount of time. This causes the flight plan to also be shifted earlier the same amount, which could result in the early flight plans being shifted much earlier than the original beginning of the scenario.

2. Consider a flight that changes its computer identification number (CID) a number of times

during a scenario. Since is considered to be a single flight, each occurrence is shifted the same amount. If the first occurrence is early in the scenario, each occurrence will be shifted only a small amount. This would cause the last occurrence to also be shifted only a small amount.

The first example would cause the scenario to begin earlier in time and the second example would cause the scenario to end at about the same time as the original. As a result the duration of the scenario would increase even though the track only data is less than the original.

Random Time Adjustment

For random time adjustment, the start time of a flight's track is modified by adding a random time increment. For example in Figure 5, the start time of the original track, T_0, is changed to T_0' by adding a random variable r, where r is randomly selected from some known frequency distribution. As implemented by CPAT, this may be from either a normal or uniform frequency distribution. In addition, the time shift values is restricted so that no flight can be shifted earlier than the base time.
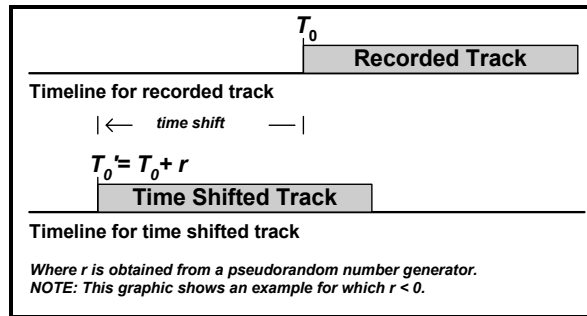
T_0

Recorded Track

Timeline for recorded track

$|\leftarrow$ **time shift** $—|$

$T_0' = T_0 + r$

Time Shifted Track

Timeline for time shifted track

Where r is obtained from a pseudorandom number generator.
NOTE: This graphic shows an example for which r < 0.

Figure 5: Random Time Adjustment

For similar reasons as those illustrated in the time-compressed scenario, the duration of a scenario that has random time adjustment will also generally be longer than that of the original scenario.

Time Shifting Using a Genetic Algorithm

A genetic algorithm is a stochastic process that is a special case of a class of algorithms called Random Heuristic Search algorithms [7]. The feasibility of its use for time shifting scenarios was documented in [8] and its implementation by CPAT was documented in [9]. In this implementation the genetic algorithm finds a set of time shift values – one for each flight in the scenario – so that the distribution of the aircraft-to-aircraft encounters meet user specified constraints. These constraints are the number of encounters in the scenario and the distribution of encounters within four

primary encounter parameters. These are: the horizontal separation at closest approach, the vertical separation at closest approach, the encounter angle at closest approach, and vertical phase of flight at closest approach (where vertical phase of flight can be that both aircraft are in level flight, that one aircraft is in level flight and the other is either descending or climbing, and that both aircraft are either descending or climbing).

As implemented by CPAT, the genetic algorithm does not impose the base time restriction found in the time compression and random time adjustment techniques. Therefore flights early in the scenario may be shifted so that their first track point is earlier than the base time depending on the frequency distribution parameters selected by the user. This causes scenarios generated by the genetic algorithm to have a longer duration that the original scenario.

Comparison of Time Shifting Techniques

To compare these three time shifting techniques a nominal (non time-shifted) scenario was generated. Then the following time-shifted scenarios were generated:

- Time-compressed only
- Random time adjustment only
- Time-compressed and random time adjustment
- Genetic algorithm

As determined earlier, 23,179 (N_r) encounters are needed in our sample application. Each of these time-shifted scenarios will be evaluated in terms of the quantity of total scenario hours needed to achieve this encounter count. Thus, each scenario will have a total encounter count and duration. Using Equation 8, an estimate of the total scenario hours (S) required for each method was calculated.

$$S = \left(\frac{N_r}{N_s}\right) D_s \qquad (8)$$

where

N_r = required total encounter count
N_s = the scenario's encounter count
D_s = the scenario duration

The results are presented in Tables 1 and 2 and are discussed in the following sections.

Table 1: Encounter Parameter Distributions

	Required Sample Size	Non Time-Shifted	Time-Shifted			
		Nominal	Time Comp	Random Time Adj	Time Comp and Random Time Adj	Genetic Algorithm
Number of Encounters	23179	5544	7151	5804	6318	8049
Horizontal Sep: 0 to 5 nm	3788	906 16.34%	1267 17.72%	1079 18.59%	1066 16.87%	1362 16.92%
Horizontal Sep: 5 to 10 nm	3934	941 16.97%	1309 18.31%	1057 18.21%	1238 19.59%	1406 17.47%
Horizontal Sep: 10 to 15 nm	5076	1214 21.90%	1437 20.10%	1152 19.85%	1326 20.99%	1727 21.46%
Horizontal Sep: 15 to 20 nm	5489	1313 23.68%	1619 22.64%	1301 22.42%	1424 22.54%	1869 23.22%
Horizontal Sep: 20 to 25 nm	4892	1170 21.10%	1519 21.24%	1215 20.93%	1264 20.01%	1685 20.93%
Vertical Sep: 0 to 1000 ft	5657	1353 24.40%	1948 27.24%	1775 30.58%	1814 28.71%	2115 26.28%
Vertical Sep: 1000 to 2000 ft	2007	480 8.66%	556 7.78%	438 7.55%	448 7.09%	696 8.65%
Vertical Sep: 2000 to 3000 ft	10452	2500 45.09%	3122 43.66%	2369 40.82%	2672 42.29%	3486 43.31%
Vertical Sep: 3000 to 4000 ft	1472	352 6.35%	408 5.71%	342 5.89%	363 5.75%	530 6.58%
Vertical Sep: 4000 to 5000 ft	3591	859 15.49%	1117 15.62%	880 15.16%	1021 16.16%	1222 15.18%
Enc Angle: 0° to 30°	3458	827 14.92%	1075 15.03%	832 14.33%	878 13.90%	1267 15.74%
Enc Angle: 30° to 60°	2860	684 12.34%	924 12.92%	749 12.90%	821 12.99%	972 12.08%
Enc Angle: 60° to 90°	2985	714 12.88%	945 13.21%	771 13.28%	866 13.71%	1007 12.51%
Enc Angle: 90° to 120°	3487	834 15.04%	1035 14.47%	888 15.30%	956 15.13%	1202 14.93%
Enc Angle: 120° to 150°	4277	1023 18.45%	1277 17.86%	1022 17.61%	1157 18.31%	1486 18.46%
Enc Angle: 150° to 180°	6112	1462 26.37%	1895 26.50%	1542 26.57%	1640 25.96%	2115 26.28%
Level-Level	13684	3273 59.04%	4205 58.80%	3343 57.60%	3709 58.71%	4323 53.71%
Level-Transitioning	7509	1796 32.40%	2368 33.11%	1928 33.22%	2060 32.61%	2843 35.32%
Transitioning-Transitioning	1986	475 8.57%	578 8.08%	533 9.18%	549 8.69%	883 10.97%

Table 2: Scenario Duration Required

	Non Time-Shifted	Time-Shifted			
	Nominal	Time Comp	Random Time Adj	Time Comp and Random Time Adj	Genetic Algorithm
Scenario Duration	14343 s 4.0 h	16750 s 4.7 h	17375 s 4.8 h	16411 s 4.6 h	17793 s 4.9 h
Track Only Duration	14340 s 4.0 h	14130 s 3.9 h	14347 s 4.0 h	13819 s 3.8 h	17671 s 4.9 h
Required Duration	60017 s 16.7 h	54292 s 15.1 h	69389 s 19.3 h	60207 s 16.7 h	51239 s 14.2 h

Scenarios are also evaluated by their distribution of several primary encounter properties. Each of the four scenarios is compared to the nominal scenario. As listed in Table 1, the encounter properties include five bins of minimum horizontal separation (e.g. 10 to 15 nautical miles), five bins of minimum vertical separation, six encounter angle bins, and three bins for the vertical phase of flight during the encounter (level-level, level-transitioning, and transitioning-transitioning). It is important that the scenarios have a proportional number of encounters in each of these bins as compared to the nominal scenario. For example, it would not be valid if a scenario achieved a high encounter count by producing many in-trail encounters (encounter angles between 0° to 30°). In-trail encounters are more sensitive to errors in speed and will lead to larger False Alert predictions by a CP. Since these properties can influence a CP's accuracy performance, all time-shifted scenarios must have encounter distributions proportional to the nominal (non time-shifted) scenario.

Nominal Scenario

The nominal scenario used for this comparison is based on data recorded at the Memphis ARTCC on October 11th, 2000.

As shown in Table 2, the duration of this scenario is about four hours (14343 seconds ≈ 4.0 hours). Since this data is based on recorded data the duration of the track-only portion of the scenario is almost the same (14340 seconds ≈ 4.0 hours). Without using time compression for the sample application it would require 16.7 scenario hours to obtain the required encounters (N_r). Table 1 also lists the scenario's quantity of encounters per encounter property bin. For example, the nominal scenario had 1,214 encounters with minimum horizontal separations between 10 to 15 nautical miles. This equates to 21.9% of the total encounters found in the scenario.

Based on the required number of encounters (N_r = 23,179) discussed earlier, Table 1 also presents each bin's target value of encounters. For example, for the same bin of minimum horizontal separation between 10 to 15 nautical miles, there are 5,076 encounters required. The bin is exactly 21.9% of the total required encounters. Therefore, the required sample size column in Table 1 lists the required quantity of encounters per bin with the same distribution as the nominal scenario.

Time Compression Only Scenario

The first scenario was only compressed in time. The compression multiplier used for this compression was 0.75. Since this was a four-hour scenario, flights near

the end of this scenario were moved almost one hour earlier in time. This one-hour maximum was chosen for this comparison because CPAT had determined through other research that a flight can be time-shifted up to one-hour without impacting conflict probe trajectory modeler's accuracy [10].

For the reasons previously discussed, the duration of this scenario (16750 seconds = 4.7 hours) is longer than that of the nominal scenario, while the track-only portion of the scenario is slightly less (14130 seconds ≈ 3.9 hours). The number of scenario hours required when using time compression for this sample application is 16.7 hours.

The time-compressed scenario's distribution of encounter properties was reasonably matched to the nominal. The largest deviation was about 3% from the nominal for the minimum vertical separation bin between 0 and 1000 feet.

Random Time Adjustment Only Scenario

The next scenario used random time adjustment. For this scenario each flight was randomly time-shifted earlier in time using a uniform frequency distribution. This random selection was done so that no flight was moved earlier in time by more than one hour and so that no flight was moved later in time.

The total duration of this scenario (17375 seconds ≈ 4.8 hours) is even greater than the time-compressed only scenario, while the track-only portion is about the same (14347 seconds ≈ 4.0 hours) as the nominal scenario. The number of scenario hours required using random time adjustment only for this sample application is 19.3 hours.

The random time-adjusted scenario's distribution of encounter properties also reasonably matched the nominal scenario. However, a few of the vertical separation bins were as high as 6.2% different than the nominal. The difference was still reasonable yet more properties deviated than the time-compressed scenario.

Time-Compressed and Random Time Adjustment Scenario

Both time compression and random time adjustment were used for the next scenario. For this scenario the flights were first compressed using a compression multiplier of 0.875, which caused no flights to be moved earlier in time by more than one-half hour. Then a random time adjustment was done using a uniform frequency distribution so that the flights could be moved earlier in time by no more than 1800 seconds. This ensured that all flights were moved

earlier than their original time, but no flight was moved earlier by more than one hour.

The total duration of this scenario is less than when the individual techniques were applied (16411 seconds ≈ 4.6 hours). For this example the duration of the track only portion is less than that of the nominal (13819 seconds ≈ 3.8 hours). The number of scenario hours required using a combination of time compression and random time adjustment for this sample application is 16.7 hours.

The time-compressed and randomly time-adjusted combined scenario had reasonably matching encounter properties as well. It's deviation from the nominal was somewhere in between the time-compressed scenario and the randomly time-adjusted version. For example, the same minimum vertical separation bin of 0 to 1000 feet was 4% larger than the nominal, which is between the 3 and 6% deviations of the respective time-compressed and randomly time-adjusted scenarios.

Genetic Algorithm Scenario

Some of the input parameters for the genetic algorithm are summarized in Table 3. Detailed information about these input parameters is provided in [9]. These particular lower bounds were selected in order to derive time shift values that would modify the original scenario so that it would attain one-third of the required encounters specified in the sample application. The upper values were selected to be 10% higher. The genetic algorithm found a solution with a fitness value of 0.99, where a fitness of 1.0 indicates that all constraints were met.

The total duration of the scenario based on this solution is longer than the original scenario (17793 seconds ≈ 4.9 hours), with a similar track only portion (17671 seconds ≈ 4.9 hours). The number of scenario hours required using the genetic algorithm for this sample application is 14.2 hours. The genetic algorithm produces a scenario with the least required scenario hours of all three previous techniques.

Since the genetic algorithm explicitly attempts to fit the various encounter bins, in general its scenario most closely matches that of the nominal compared to the other three scenarios. With the exception of the vertical phase of flight, all the bins are matched within 2% of the nominal scenario. The vertical phase of flight bins had deviations as large as 5%. This is easily explained: the current implementation of CPAT's genetic algorithm approximates the vertical phase of flight by taking it only at the closest point of approach (time point of minimum horizontal

separation). A future implementation may be designed to determine if each flight was transitioning at any point during the encounter.

Table 3: Selected Genetic Algorithm Input

Input Parameter	Low Bound	High Bound
Desired number of encounters	7726	8499
Min Horz Sep: 0 to 5 nm	1263	1389
Min Horz Sep: 5 to 10 nm	1311	1442
Min Horz Sep: 10 to 15 nm	1692	1861
Min Horz Sep: 10 to 20 nm	1830	2013
Min Horz Sep: 20 to 25 nm	1631	1794
Min Vert Sep: 0 to 1000 ft	1886	2075
Min Vert Sep: 1000 to 2000 ft	669	736
Min Vert Sep: 2000 to 3000 ft	3484	3832
Min Vert Sep: 3000 to 4000 ft	491	540
Min Vert Sep: 4000 to 5000 ft	1197	1317
Encounter Angle: 0° to 30°	1153	1268
Encounter Angle: 30° to 60°	953	1048
Encounter Angle: 60° to 90°	995	1095
Encounter Angle: 90° to 120°	1162	1278
Encounter Angle: 120° to 150°	1426	1569
Encounter Angle: 150° to 180°	2037	2241
Level - Level	4561	5017
Level - Transitioning	2503	2753
Transitioning - Transitioning	662	728

Summary

Building upon the fundamental principles discussed in [5], this paper introduced a method to estimate the number of aircraft encounters and the scenario hours required to test a CP. A hypothesis testing technique was developed that balanced competing α and β risks for the CP developer and user. This was demonstrated on testing a CP's probability of False Alerts.

Next, the concept of time shifting recorded air traffic was presented. This involves moving aircraft flights temporally using three different techniques: time compression, random time adjustment, and a genetic algorithm. These techniques were applied on a sample of recorded air traffic from Memphis ARTCC, producing four scenarios, which included a time-compressed scenario, randomly time-adjusted scenario, a combined time-compressed and randomly time-adjusted scenario, and a genetic algorithm generated scenario. The scenarios were evaluated against a non time-shifted or nominal scenario to meet a required number of aircraft encounters (based on the hypothesis test discussed). Each scenario was also evaluated on how well it matched the distribution of encounter properties of the nominal scenario. The encounter properties examined are listed in Table 1 (e.g., minimum horizontal

separation). The time compression technique performed well, but the overall best technique was the genetic algorithm. It explicitly was designed to achieve a specified quantity of encounters as listed in Table 3 and resulted in a scenario requiring 15% less scenario hours.

An effective application of these techniques would be a developer's regression testing of CP software releases or versions. A scenario or set of scenarios can be base lined with an existing software release. Future releases are then hypothesis tested against the performance of the base line release (acting as the requirement). If the current CP software release fails the test, it is examined for software problems, corrected, and retested. Furthermore, with the proper support tools, the actual Missed and False Alert events can be examined in detail to aid the developer in making the proper corrections to the current CP release. A time-shifted air traffic scenario with a significant number of conflicts and encounters can be used repeatedly, maybe for several years depending on the program. Therefore, a reduction in the required scenario hours by only a few hours (like presented in the genetic algorithm scenario) can amount to many hours of labor and lab utilization savings over several years.

Future research will expand this work to include the evaluation of time shifting methods of aircraft encounters and concurrent aircraft-to-aircraft conflicts (loss of legal separation) as presented in [11]. Thus, both the False Alert and Missed Alert probabilities could be evaluated with the time-shifted scenarios. Also, applications of the quantitative measurement methodology and the scenario generation will be explored.

References

[1] Kazunas, Stephen (2002), "The Application of Statistical Hypothesis Methods to Quantitative Requirements Testing," *47th Annual Air Traffic Control Association Conference Proceedings*, Washington, DC, November.

[2] Paglione, Mike M., Mary Lee Cale, Hollis F. Ryan (1999), "Generic Metrics for the Estimation of the Prediction Accuracy of Aircraft to Aircraft Conflicts by a Strategic Conflict Probe Tool," *Air Traffic Control Quarterly*, Vol. 7, No. 3, pp. 147-165.

[3] Bilimoria, K.D., 2001, "Methodology for the Performance Evaluation of a Conflict Probe,"

Journal of Guidance, Control, and Dynamics, Vol. 24, No. 3, pp. 444-451.

[4] Arthur, William C., and Dr. Michael P. McLaughlin (1998), "User Request Evaluation Tool (URET) Interfacility Conflict Probe Performance Assessment," *2nd USA/Europe Air Traffic Management R&D Seminar*, Orlando, FL, December.

[5] Devore, Jay L. (2000), *Probability and Statistics for Engineering and the Sciences*, 5th Edition, Brooks/Cole, Belmont, CA.

[6] Oaks, Robert D., and Mike Paglione (2001), "Generation of Realistic Air Traffic Scenarios Based on Recorded Field Data," *46th Annual Air Traffic Control Association Conference Proceedings*, Washington, DC, November.

[7] Vose, Michael D. (1999), *The Simple Genetic Algorithm: Foundations and Theory*, Cambridge, MA, The MIT Press.

[8] Oaks, R. D. (2002), "A Study on the Feasibility of Using a Genetic Algorithm to Generate Realistic Air Traffic Scenarios Based on Recorded Field Data," Paper No. 2002-4767, *AIAA Guidance, Navigation, and Control Conference*, Monterey, CA, August.

[9] Oaks, R. D., and M. M. Paglione (2002), "Generation of Realistic Air Traffic Scenarios Using a Genetic Algorithm," *21st Digital Avionics Systems Conference: Air Traffic Management for Commercial and Military Systems*, Irvine, CA, October.

[10] Paglione, Mike M., Robert D. Oaks, Dr. Hollis F. Ryan, J. Scott Summerill (1999), "User Request Evaluation Tool Daily Use Time-Shifted Trajectory Accuracy Degradation Study," December 15, www.acy.tc.tatca.faa.gov/cpat/docs/index.shtml.

[11] Paglione, Mike M., Robert D. Oaks, Karl D. Bilimoria, (2003), "Methodology for Generating Conflict Scenarios by Time Shifting Recorded Traffic Data," (Submittal) *5th USA/Europe Air Traffic Management R&D Seminar*, Budapest, Hungary, June.